TECHNICAL RESEARCH REPORT

Combined Compression and Classification with Learning Vector Quantization

by J. Baras, S. Dey

T.R. 98-26

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
1998		**-**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Combined Compression and Classification with Learning Vector Quantization	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Office of Naval Research,One Liberty Center,875 North Randolph Street Suite 1425,Arlington,VA,22203-1995**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT
see report

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**		**20**	

Combined Compression and Classification with Learning Vector Quantization *

John S. Baras † Subhrakanti Dey ‡

June 15, 1998

Abstract

Combined compression and classification problems are becoming increasingly important in many applications with large amounts of sensory data and large sets of classes. These applications range from aided target recognition (ATR), to medical diagnosis, to speech recognition, to fault detection and identification in manufacturing systems. In this paper, we develop and analyze a learning vector quantization (LVQ) based algorithm for the combined compression and classification problem. We show convergence of the algorithm using techniques from stochastic approximation, namely, the ODE method. We illustrate the performance of our algorithm with some examples.

Index Terms- **Learning vector quantization, classification, stochastic approximation, compression, non-parametric**

1 Introduction

Quite often in applications, we are faced with the problem of classifying signals (or objects) from vast amounts of noisy data. Equally often, the number of different distinct signals (classes) that we have in the problem may be quite large. If we could compress each observation (observed signal) significantly without distorting or annihilating the most significant features used for classification, we can achieve significant advantages in two directions:

(i) We can reduce significantly the memory required for storing both the on-line and class model data;

(ii) We can increase significantly the speed of searching and matching that is essential in any classification problem.

Furthermore, performing classification on compressed data can result in better classification, due to the fact that compression (done correctly) can reduce the noise more than the signal [1]. For all these

*Research supported by ONR contract 01-5-28834 under the MURI Center for Auditory and Acoustics Research, by NSF grant 01-5-23422 and by the Lockheed Martin Chair in Systems Engineering.

†Department of Electrical Engineering and Institute for Systems Research, University of Maryland, College Park, MD 20742

‡Institute for Systems Research, University of Maryland, College Park, MD 20742

reasons, it is important to develop methods and algorithms to perform classification of compressed data, or to analyze jointly the problem of compression and classification. This area has attracted recently more interest due to the increased number of applications requiring such algorithms. In [2] and [3], vector quantization methods have been used for minimizing both the distortion of compressed images and errors on classifying their pixel blocks.

There is yet another significant advantage in investigating the problem of combined compression and classification. If such a framework is developed, we can then analyze progressive classification schemes, which offer significant advantages for both memory savings and for speeding up searching and matching. Progressive classification uses very compressed representations of the signals at first to perform many simple (and therefore fast) matching tests, and then progressively perform fewer but more complex (and therefore slower) matching tests, as needed for classification. Thus, compression becomes an indispensable component in such schemes, and in particular variable rate (and therefore resolution) compression. In the last four years, we have analyzed such progressive classification schemes on a variety of problems with substantial success. The structure of the algorithms we have developed has remained fairly stable, regardless of the particular application. This structure consists of a multiresolution preprocessor followed by a tree-structured classifier as the postprocessor. Sometimes a nonlinear feature extraction component needs to be placed between these two components. Often the postprocessor incorporates learning.

To date, we have utilized wavelets as the multiresolution preprocessor and Tree-structured-vector-quantization (TSVQ) as the clustering postprocessor. We have applied the resulting WTSVQ algorithm to various ATR problems based on radar [4] [5] [6], ISAR and face recognition problems [7]. We have established similar results on ATR based on FLIR using polygonization of object silhouettes [8] [9] as the multiresolution preprocessor. Incorporation of compression into these algorithms is part of our current research.

As a first step towards developing a progressive classification scheme with compression, we need to develop an algorithm for combined compression and classification at a fixed resolution. As opposed to the algorithm described in [3] that achieves this with posteriori estimation of the probability models underlying the different classes of signals, our goal is to develop an algorithm that is nonparametric, in the sense that it does not use estimates of probability distributions of the underlying sources generating the data. In this paper, we achieve that goal by using a variation of Learning Vector Quantization (LVQ), that cleverly takes into account the distortion present. Note that LVQ as described in [10], although designed to perform classification, automatically achieves some compression as a byproduct since it is inherently a vector quantization algorithm. However, our algorithm is designed to obtain a systematic trade-off between its compression and classification performances by minimizing a linear combination of the compression error (measured by average distortion) and classification error (measured by Bayes risk) with a variation of LVQ based on a stochastic approximation scheme. The convergence analysis of this algorithm essentially follows similar techniques as presented in [11] and as used in [12]. However, our treatment is considerably simpler since to start with, we recognize that the algorithm is a special class

of the Robbins-Monro algorithm.

In Section 2, we describe the LVQ-based algorithm for combined compression and classification. In Sections 2.1 and 2.2, we provide analysis and convergence of the algorithm using stochastic approximation techniques and the so-called ODE method. In Section 3, we provide simulation results of the performance of the algorithm for some typical problems. Section 4 presents some concluding remarks.

2 Combined compression and classification with learning vector quantization

Learning vector quantization (LVQ) introduced in [13] is a nonparametric method of pattern classification. As opposed to the parametric methods, this method does not attempt to obtain a-posteriori estimates of the underlying probability models of the different patterns that generate the data to be classified. It simply uses a set of training data for which the classes are known in a supervised learning algorithm to divide the data space into a number of Voronoi cells represented by the corresponding Voronoi vectors and their associated class decisions. Using the training vectors, these Voronoi vectors are updated iteratively until they converge. The algorithm involves three main steps:

1. Find out which Voronoi cell a given training vector belongs to by the nearest-neighbor rule.
2. If the decision of the training vector coincides with that of the Voronoi vector of this particular cell, move the Voronoi vector towards the training vector, else, move it away from the training vector. All the other Voronoi vectors are not changed.
3. Obtain the next training vector and perform the first two steps.

This process is usually carried out in multiple passes of the finite set of the training vectors. A detailed description of this algorithm with a preliminary analysis of its convergence properties using stochastic approximation techniques of [11] has been given in [12]. It has also been indicated in [12] that as the number of training vectors goes to infinity, the classification error achieved by the LVQ algorithm approaches the optimal Bayes' error. Although its primary goal is to classify the data into different patterns, the LVQ algorithm compresses the data in the process into a codebook of the size equal to the number of the Voronoi cells where each Voronoi vector represents the code for all the vectors belonging to that cell.

In what follows, we present a simple variation of the LVQ algorithm in [12], that achieves a task of combined compression and classification. We present a convergence analysis of this algorithm much along the lines of [12]. However, we present a simpler analysis by recognizing that the algorithm is a special case of the Robbins Monro algorithm. Also, simulation results show that as a certain parameter is increased, the compression error gradually decreases compared to the error achieved by the standard LVQ (represented by the value zero of this parameter).

In the next subsection, we introduce the notations and describe the algorithm.

Algorithm for combined compression and classification

Consider a complete probability space $(\Omega, \mathcal{F}, P)$. Let $X_l \in \mathbb{R}^d,\ l = 1, 2, \ldots, N$ represent the training vectors defined on this space, generated by either of the two patterns 1 or 2. The a priori probabilities of the two patterns are π_1 and π_2 respectively and the corresponding pattern densities are $p_1(x)$ and $p_2(x)$ respectively such that

$$P(X_l \in B) = \pi_1 \int_B p_1(x)dx + \pi_2 \int_B p_2(x)dx \tag{1}$$

We also assume that X_l is independent of $X_j, j \neq l$.

The Voronoi vectors are represented by $\theta_i \in \mathbb{R}^d,\ i = 1, 2, \ldots, K$ and the corresponding Voronoi cells are represented by V_{θ_i}. Let the decision associated with the training vector X_l be represented by d_{X_l} and that of the cell V_{θ_i} by d_{θ_i}, where $d_{X_l}, d_{\theta_i} \in \{1, 2\}$.

Consider a non-increasing sequence of positive real numbers $\epsilon_n,\ n = 1, 2, \ldots$, such that

Assumption 2.1 $\sum_{n=1}^{\infty} \epsilon_n = \infty$

Consider also a distance function $\rho(\theta, x)$ which satisfies the following assumptions:

Assumption 2.2 *$\rho(\theta, x)$ is a twice continuously differentiable function of θ and x and for every fixed $x \in \mathbb{R}^d$, it is a convex function of θ.*

Assumption 2.3 *For any fixed x, if $\theta(k) \to \infty$, as $k \to \infty$, then $\rho(\theta(k), x) \to \infty$.*

Assumption 2.4 *For every compact $Q \in \mathbb{R}^d$, there exist constants C_1 and q_1 such that for all $\theta \in Q$,*

$$\mid \nabla_\theta \rho(\theta, x) \mid < C_1(1 + \mid x \mid^{q_1}) \tag{2}$$

An example of a function which satisfies the properties above is $\rho(\theta, x) = ||\theta - x||^2$ where $||.||$ is the Eucledian distance between two vectors. In our implementation of the algorithm, we use this distance function although for the sake of generality in the analysis, we would refer to it in its general form $\rho(\theta, x)$.

Define further the following quantities:

Definition 2.1

$$\gamma(d_{X_{n+1}}, d_{\theta_i(n)}, X_{n+1}, \Theta(n)) = -1_{X_{n+1} \in V_{\theta_i(n)}} (1_{d_{X_{n+1}} = d_{\theta_i(n)}} - 1_{d_{X_{n+1}} \neq d_{\theta_i(n)}}) \tag{3}$$

where $\Theta(n) = (\theta_1(n), \ldots, \theta_K(n))'$ and $\theta_i(n)$ is the n-th iterate of θ_i, $n \geq 0$. Also 1_A is the indicator function that takes the value 1 if A is true and 0 otherwise.

Definition 2.2

$$\begin{aligned} g_i(\Theta(n); N) &= 1\ if\ \frac{1}{N} \sum_{j=1}^{N} 1_{X_j \in V_{\theta_i(n)}} 1_{d_{X_j} = 1} > \frac{1}{N} \sum_{j=1}^{N} 1_{X_j \in V_{\theta_i(n)}} 1_{d_{X_j} = 2} \\ &= 2\ otherwise \end{aligned} \tag{4}$$

Remark 2.1 Note that $g_i(\Theta(n); N)$ above denotes the decision associated with the i-th cell according to the majority vote rule.

With the above definitions and assumptions, we can now write the following multi-pass combined compression and classification algorithm for $\lambda \geq 0$,

1. *Initialization*: The algorithm is initialized with $\Theta(0)$ usually found by running a vector quantization algorithm, e.g., LBG [14] algorithm over the set of training vectors.
2. $n = 0$.
3. *Assigning the training vectors to their respective cells*: Find $i_l = \text{argmin}_m \, ||\theta_m(n) - X_l||^2, \; l = 1, 2, \ldots, N$, then X_l belongs to $V_{\theta_{i_l}(n)}$.
4. *Cell decisions*: Calculate $g_i(\Theta(n); N), \; i = 1, 2, \ldots, K$.
5. *Updating the Voronoi vectors*: For $i \in \{1, 2, \ldots, K\}$,

$$\theta_i(n+1) = \theta_i(n) + \epsilon_{n+1}(-\lambda 1_{X_{n+1} \in V_{\theta_i(n)}} + \gamma(d_{X_{n+1}}, g_i(\Theta(n); N), X_{n+1}, \Theta(n)))\nabla_\theta \rho(\theta, X_{n+1}) \mid_{\theta=\theta_i(n)} \tag{5}$$

6. $n \leftarrow n + 1$.
7. If $n < N$, repeat Steps 3-6. If $n = N$, repeat Steps 3-4.

The above algorithm can be executed for multiple passes over the same training set (in case the size of the training set is small) by using the values $\Theta(N)$ from the m-th pass to initialize the algorithm for pass = $m + 1$ until $m = M$ where M is the maximum number of passes.

Remark 2.2 Note that Step 5, i.e., updating of the Voronoi vectors can be written in the following simplified manner:
If $X_{n+1} \in V_{\theta_i(n)}$, then

$$\begin{aligned} \theta_i(n+1) &= \theta_i(n) + \epsilon_{n+1}(-\lambda - 1)\nabla_\theta \rho(\theta, X_{n+1}) \mid_{\theta=\theta_i(n)} \; if \; d_{X_{n+1}} = g_i(\Theta(n); N) \\ &= \theta_i(n) + \epsilon_{n+1}(-\lambda + 1)\nabla_\theta \rho(\theta, X_{n+1}) \mid_{\theta=\theta_i(n)} \; if \; d_{X_{n+1}} \neq g_i(\Theta(n); N) \end{aligned} \tag{6}$$

For $j \neq i$, $\theta_j(n+1) = \theta_j(n)$.

Remark 2.3 Note that for $\lambda = 0$, the above algorithm becomes the modified LVQ algorithm resulting in better convergence properties as reported in [12].

2.1 Analysis of the combined compression and classification algorithm

In this subsection, we present the analysis of the above algorithm using the "mean ODE" method of [11].

Denote the vectors

$$h(\Theta(n)) = (h_1(\Theta(n)), \ldots, h_K(\Theta(n))'$$

and

$$H(\Theta(n), X_{n+1}) = (H_1(\Theta(n), X_{n+1}), \ldots, H_K(\Theta(n), X_{n+1}))'$$

where

$$H_i(\Theta(n), X_{n+1}) = (-\lambda 1_{X_{n+1} \in V_{\theta_i(n)}} + \gamma(d_{X_{n+1}}, g_i(\Theta(n); N), X_{n+1}, \Theta(n)))\nabla_\theta \rho(\theta, X_{n+1}) \mid_{\theta=\theta_i(n)} \tag{7}$$

and $h_i(\Theta(n)),\ i = 1, 2, \ldots, K$ is defined in Definition 2.4. Note that one can write the above algorithm (5) in the following manner:

$$\Theta(n+1) = \Theta(n) + \epsilon_{n+1} H(\Theta(n), X_{n+1}),\ n \geq 0 \tag{8}$$

Note that this is a special case of the general stochastic approximation algorithm of [11], quoted in Section 2, [12].

Define

$$\begin{aligned} p(x) &= p_1(x)\pi_1 + p_2(x)\pi_2 \\ q(x) &= p_2(x)\pi_2 - p_1(x)\pi_1 \end{aligned} \tag{9}$$

Due to the assumption that $\{X_l\},\ l = 1, 2, \ldots,$ is a sequence of *i.i.d.* random vectors and the fact that they are distributed independently of $\Theta(l)$, the transition probability function $\Pi_{\Theta(n)}(A, X_n) \stackrel{\triangle}{=} P(X_{n+1} \in A \mid \mathcal{F}_n)$ is given by $\mu(A) = \int_A p(x)dx$, where $\mathcal{F}_n \stackrel{\triangle}{=} \sigma\{\Theta(0), X_0, \ldots, \Theta(n), X_n\}$. This makes the above algorithm a special case of the Robbins-Monro algorithm with the transition probability function being independent of $\Theta(n)$.

Now, we introduce the following definitions:

Definition 2.3

$$\bar{\gamma_i}(\Theta(n); N) = sign\left(\frac{1}{N}\sum_{j=1}^{N} 1_{X_j \in V_{\theta_i(n)}}(1_{d_{X_j}=2} - 1_{d_{X_j}=1})\right) \tag{10}$$

Remark 2.4 Note that $\gamma_i(\Theta(n); N) = 1$ if $g_i(\Theta(n); N) = 2$ and -1 otherwise.

Definition 2.4

$$h_i(\Theta) = -\int_{V_{\theta_i}} [\bar{\gamma_i}(\Theta; N)q(x) + \lambda p(x)]\nabla_\theta \rho(\theta, x)_{|\theta=\theta_i} dx,\ i = 1, 2, \ldots, K \tag{11}$$

One can now prove the following Lemma:

Lemma 2.1

$$H_i(\Theta(n), X_{n+1}) = h_i(\Theta(n)) + \xi_i(n),\ i = 1, 2, \ldots, K \tag{12}$$

where $\{\xi_i(n)\}$ is a $\mathcal{F}_n$-adapted martingale difference sequence such that

$$h_i(\Theta(n)) = E_a[H_i(\Theta(n), X_{n+1}) \mid \mathcal{F}_n],\ \forall i \tag{13}$$

Here, E_a denotes expectation under P_a where P_a denotes the probability distribution for $\{X_n, \Theta(n)\},\ n \geq 0$ where $\Theta(0) = a$. Note that since $\{X_n\}$ is a sequence of *i.i.d.* random vectors, P_a is independent of $X_0 = x$.

We write the mean ODE associated with (8) as

$$\dot{\Theta} = h(\Theta),\ \Theta(0) = a \tag{14}$$

where

$$h_i(\Theta) = \lim_{n\to\infty} E_a[H_i(\Theta, X_{n+1})|\mathcal{F}_n] = \int H_i(\Theta, x)p(x)dx \tag{15}$$

since in this case $\{X_n\}$ is a sequence of *i.i.d.* random variables where $P(X_{n+1} \in A \mid \mathcal{F}_n)$ is independent of $\Theta(k),\ k \leq n$.

It is hard to establish a convergence result for general $h(\Theta)$ and often it is assumed that (14) has an attractor Θ^*, whose domain of attraction is given by D^*. If Q is a compact subset of D^* and $\Theta(0) = a\ \in Q$, one can show that for any $\delta > 0$,

$$P\{\max_n ||\Theta(n) - \Theta(a, t_n)|| > \delta\} < C(\alpha, Q) \sum_n \epsilon_n^\alpha \tag{16}$$

where $t_n = \sum_{i=1}^n \epsilon_i$ and $\Theta(a, t_n)$ is the solution to (14) for $t = t_n$, and $C(\alpha, Q)$ is a constant dependent on α and Q (see Theorem 4, page 45, [11]). Here, obviously, we have assumed Assumption 2.1.

One could also derive the following corollary (see Corollary 6, page 46, [11]), which says that under the assumptions (16) is true, for the set of trajectories $\{\Theta(n)\}$ that visit Q infinitely often, we have

$$\Theta(n) \to \Theta^*,\ P_a - a.s. \tag{17}$$

$$P\{\limsup_{n\to\infty} ||\Theta(n) - \Theta(a, t_n)|| > \delta\} = 0 \tag{18}$$

However, there is no general theory which gives conditions under which $P(\Theta(n) \in Q\ infinitely often\ ) = 1$ is satisfied [11].

Note that for a complete theory, it is essential to prove that the desired points of convergence θ^* are indeed the stable equilibrium points of (14). One way to do this is to find a potential function $J(\Theta)$, if it exists, such that $h_i(\Theta) = -\nabla_{\theta_i} J(\Theta)$. Then one can apply results from Lyapunov stability to establish results for stable equilibrium by studying the local minima of $J(.)$ and their domains of attraction. Although, we refrain from such pursuits for the time being, we do notice that (see [12]) as $N \to \infty$, $\bar{\gamma}_i(\Theta; N) \to sign(\int_{V_{\theta_i}} q(x)dx)$ and using the mean value theorem when the size of each Voronoi cell is small, one can write that $h_i(\Theta)$ is approximately equal to

$$\bar{h}_i(\Theta) \approx -\int_{V_{\theta_i}} \nabla_\theta \rho(\theta, x)_{|_{\theta=\theta_i}} (|q(x)| + \lambda p(x))dx \tag{19}$$

which is the negative gradient of the cost function

$$\bar{J}(\Theta) = \sum_{i=1}^K \int_{V_{\theta_i}} \rho(\theta_i, x)(|q(x)| + \lambda p(x))dx \tag{20}$$

For those readers who are more oriented towards intuitive reasoning, we comment here that this was indeed the inspiration of obtaining the combined compression and classification algorithm given

above. The reason for this intuition is that as indicated in [12], for the LVQ algorithm, the first part of the integrand in (20) converges to the optimal Bayes cost when the number of Voronoi vectors tends to infinity. Details of this analysis can be found in [10]. The second part is clearly the average distortion.

2.2 Convergence analysis of the combined compression and classification algorithm

The convergence analysis for a class of learning vector quantization algorithm was presented in [12] following the analysis in [11] (see Part II- Chapter 1). However, as we noted before that the algorithm under investigation is a special case of the Robbins-Monro algorithm, where the transition probability function is independent of Θ, we can simplify the set of assumptions needed greatly. In particular, the assumptions described as A.4 in [11], pp. 216, become trivial and follow from the single assumption that $h(\Theta)$ is locally Lipschitz. In this subsection, we obtain an upper bound on the L_q estimate of a "fluctuation" term to be introduced shortly, for $q > 2$. We will provide a simpler local bound later on for $q = 2$.

Consider again the algorithm:

$$\Theta(n+1) = \Theta(n) + \epsilon_{n+1} H(\Theta(n), X_{n+1}),\ n \geq 0 \tag{21}$$

Before we introduce the set of assumptions needed for the analysis of our algorithm, for the purpose of this section, let us introduce the following notations:

Notation 2.1 *1. D is an open subset of $\mathbb{R}^d$. Q is a compact subset of D.*

2. ϕ is a C^2 function from $\mathbb{R}^d$ to $\mathbb{R}$ with bounded second derivatives, where

$$\begin{aligned} M_0(Q) &= \sup_{\Theta \in Q} |\phi(\Theta)| \\ M_1(Q) &= \sup_{\Theta \in Q} |\phi'(\Theta)| \\ M_2(Q) &= \sup_{\Theta \in Q} |\phi''(\Theta)| \\ M_2 &= \sup_{\Theta \in \mathbb{R}^d} |\phi''(\Theta)| \end{aligned} \tag{22}$$

3. There exists a $R(\phi, \Theta, \Theta^p)$ such that

$$\begin{aligned} R(\phi, \Theta, \Theta') &= \phi(\Theta') - \phi(\Theta) - \langle (\Theta' - \Theta), \phi'(\Theta) \rangle \\ |R(\phi, \Theta, \Theta')| &\leq M_2 |\Theta' - \Theta|^2,\ \forall \Theta,\ \Theta' \in \mathbb{R}^d \end{aligned} \tag{23}$$

4.

$$e_n(\phi) = \phi(\Theta(n+1)) - \phi(\Theta(n)) - \epsilon_{n+1} \langle \phi'(\Theta(n)), h(\Theta(n)) \rangle \tag{24}$$

5. For $\varepsilon > 0$,

$$\begin{aligned} \tau(Q) &= \inf(n;\ \Theta(n) \notin Q) \\ \sigma(\varepsilon) &= \inf(n \geq 1;\ |\Theta(n) - \Theta(n-1)| > \epsilon) \\ \nu(\varepsilon, Q) &= \inf(\tau(Q), \sigma(\epsilon)) \end{aligned} \tag{25}$$

6. *With* $t_0 = 0,\ t_n = \sum_{i=1}^{n} \epsilon_i$, *we define* $m(n,T) \stackrel{\triangle}{=} \inf\{k :\ k \geq n,\ \sum_{i=n}^{k} \epsilon_{i+1} \geq T\}$

Suppose Assumption 2.1 holds. Also, let us make the following additional assumptions that will be sufficient for our analysis:

Assumption 2.5 *For any compact subset* Q *of* D, *there exist constants* $\bar{C_1}, r_1$ *such that*

$$|H(\Theta, x)| \leq \bar{C_1}(1 + |x|^{r_1}) \tag{26}$$

Remark 2.5 Note that for our choice of $H(\Theta, x)$ described in the previous section, (26) is satisfied from Assumption 2.3.

Assumption 2.6 $h(\Theta) \stackrel{\triangle}{=} (h_1(\Theta), \ldots, h_k(\Theta))'$ *where* $h_i(\Theta)$ *given by (13) is locally Lipschitz.*

Remark 2.6 Note that this assumption itself is enough in our analysis and we do not need the assumptions made in [12] following [11] (Assumption (A.4), pp. 216) since they trivially follow from Assumption 2.6.

Assumption 2.7 *For any* $q \geq 1$, $\exists$ *a constant* $M < \infty$ *such that*

$$\sup_n E\{|X_n|^q I(n \leq \nu(\varepsilon, Q))\} \leq M \tag{27}$$

Remark 2.7 Since $\{X_n\}$ is a sequence of *i.i.d.* random vectors, one can simply write (27) as

$$\int_{\mathbf{R}^d} |x|^q \mu(dx) dx \leq M \tag{28}$$

Remark 2.8 One can in fact deduce from Assumptions 2.5 and 2.7 that under certain other restrictions on the distribution function $\mu(dx)$, that Assumption 2.6 holds, since in this case $\mu(dx)$ is independent of Θ (see Section 2.2.6, pp. 264-265 of [11]).

We can now present the following theorem:

Theorem 2.1 *Consider the update equation (21). Consider also (24), (25). Suppose Assumptions 2.1, 2.5, 2.6, 2.7 hold. Then, for any regular function* ϕ *with bounded second derivatives satisfying (22), any compact subset* Q *of* D, *and for all* $q > 2$ *there exist constants* $B(q)$, M_4, $\varepsilon_0 > 0$ *such that for all* $\varepsilon < \varepsilon_0$, $T > 0$, $a \in D$, *we have*

$$E_a\{\sup_{n<k\leq m(n,T)} I(k \leq \nu(\varepsilon, Q))| \sum_{i=n}^{k-1} e_i(\phi)|^q\} \leq B(q) M_1(Q) T^{\frac{q}{2}-1} \sum_{i=n+1}^{m(n,T)} \epsilon_{i+1}^{1+\frac{q}{2}} + M_4(q) T^{q-1} \sum_{i=n+1}^{m(n,T)} \epsilon_{i+1}^{1+q} \tag{29}$$

Proof: In this proof, $C_1(q), C_2(q), C_3(q), C_4(q), B(q), M_4(q)$ denote constants dependent only on q. From (24), (23) and (21), one can write

$$\begin{aligned} e_k(\phi) &= \epsilon_{k+1}\langle \phi'(\Theta(k)), (H(\Theta(k), X_{k+1}) - h(\Theta(k)))\rangle + R(\phi, \Theta(k), \Theta(k+1)) \\ &= e_k^{(1)} + e_k^{(2)} \end{aligned} \tag{30}$$

where

$$e_k^{(1)} = \epsilon_{k+1}\langle \phi'(\Theta(k)), (H(\Theta(k), X_{k+1}) - h(\Theta(k)))\rangle$$

$$e_k^{(2)} = R(\phi, \Theta(k), \Theta(k+1))$$

Note that we have

$$\begin{aligned} |\sum_{i=n}^{k-1} e_i(\phi)|^q &= |\sum_{i=n}^{k-1} e_i^{(1)} + \sum_{i=n}^{k-1} e_i^{(2)}|^q \\ &\leq [|\sum_{i=n}^{k-1} e_i^{(1)}| + |\sum_{i=n}^{k-1} e_i^{(2)}|]^q \\ &\leq 2^{q-1}[|\sum_{i=n}^{k-1} e_i^{(1)}|^q + |\sum_{i=n}^{k-1} e_i^{(2)}|^q] \end{aligned} \tag{31}$$

From now on, we write m for $m(n,T)$ and ν for $\nu(\varepsilon, Q)$ for notational simplicity. We write $S_1 = E\{\sup_{n<k\leq m} I(k \leq \nu)|\sum_{i=n}^{k-1} e_i^{(1)}|]^q\} = E\{\sup_{n<k\leq m} |\sum_{i=n}^{k-1} U_i|^q\}$ where

$$U_i = \epsilon_{i+1}\langle \phi'(\Theta(i)), (H(\Theta(i), X_{i+1}) - h(\Theta(i)))\rangle I(i+1 \leq \nu)$$

Denoting $V_i = \langle \phi'(\Theta(i)), (H(\Theta(i), X_{i+1}) - h(\Theta(i)))\rangle I(i+1 \leq \nu)$, we have $U_i = \epsilon_{i+1}V_i$.

We notice that from (13), $E(U_{i+1}|\mathcal{G}_i) = 0$.

We also observe that from (22)

$$\begin{aligned} E|V_i|^q &\leq M_1(Q)C_1(q)[E|H(\Theta(i), X_{i+1})|^q + E|h(\Theta(i))|^q] \\ &\leq M_1(Q)C_2(q)E|H(\Theta(i), X_{i+1})|^q \end{aligned} \tag{32}$$

The last inequality follows from Jensen's inequality and (13).

One can now use Assumption 2.5 and Assumption 2.7 to obtain the following upper bound:

$$E|V_i|^q \leq M_1(Q)C_3(q) \tag{33}$$

One can now apply Burkholder's inequality (see Lemma 6, pp. 294, [11]) to obtain

$$S_1 \leq C_4(q)E(\sum_{i=n}^{m-1} \epsilon_{i+1}^2 V_i^2)^{q/2} \tag{34}$$

For $q > 2$, one can further apply a result based on Holder's inequality (see Lemma 7, pp. 294, [11]) to obtain

$$\begin{aligned} S_1 &\leq C_4(q)(\sum_{i=n}^{m-1} \epsilon_{i+1})^{\frac{q}{2}-1} \sum_{i=n}^{m-1} \epsilon_{i+1}^{1+\frac{q}{2}} E|V_i|^q \\ &\leq B(q)M_1(Q)T^{\frac{q}{2}-1} \sum_{i=n}^{m-1} \epsilon_{i+1}^{1+\frac{q}{2}} \end{aligned} \tag{35}$$

We prove the following bound on $S_2 = E\{\sum_{i=n}^{m-1} |e_i^{(2)}| I(i+1 \leq \nu)\}^q$ using (23),(26) and Assumption 2.7:

$$S_2 \leq M_4(q) T^{q-1} \sum_{i=n}^{m-1} \epsilon_{i+1}^{1+q} \tag{36}$$

Combining (35), (36), we obtain (29) from (31). □

Next, we present a theorem that gives an upper bound on the L_q norm of the distance between the actual iterate $\Theta(n)$ and $\Theta(a, t_n)$ which is the solution to (14) for $t = t_n$. In other words, this result gives an upper bound on the quality of approximation by the mean trajectory represented by (14). We do not provide the proof since the result holds under the same set of assumptions as the previous theorem and the proof can be found in [11], pp. 301.

Theorem 2.2 *Consider the update equation (21) and (14). Suppose Assumptions 2.1, 2.5, 2.6, 2.7 hold. Suppose $Q_1 \subset Q_2$ are compact subsets of D, and $q > 2$. Then there exist constants $B_1(q), \bar{L}_2$ ($\bar{L}_2$ is the Lipschitz constant for h in Q_2), such that for all $T > 0$ (that satisfy the condition that for all $a \in Q_1$, all $t \leq T$, $d(\Theta(a,t), Q_2^c) \geq \delta_0 > 0$), all $\delta < \delta_0$, all $a \in Q_1$,*

$$P_a\{\sup_{n \leq m(0,T)} |\Theta(n) - \Theta(a, t_n)|^q \geq \delta\} \leq \frac{B_1(q)}{\delta^q} (1+T)^{q-1} \exp(q\bar{L}_2 T) \sum_{i=1}^{m(0,T)} \epsilon_i^{1+\frac{q}{2}} \tag{37}$$

We now present an asymptotic result without proof that states that $\Theta(n)$ asymptotically converges to a compact subset of D, based on the assumption that the mean ODE has a point of asymptotic stability Θ^* in D with domain of attraction D. We make more precise statements later. First, we introduce the following additional assumptions and notations:

Assumption 2.8 *There exists α such that $\sum \epsilon_n^\alpha < \infty$.*

Assumption 2.9 *There exists a positive function U of class C^2 on D such that $U(\Theta) \to C \leq \infty$ if $\Theta \to \partial D$ or $|\Theta| \to \infty$ and $U(\Theta) < C$ for $\Theta \in D$ satisfying*

$$\langle U'(\Theta), h(\Theta) \rangle \leq 0, \ \forall \Theta \in D \tag{38}$$

Remark 2.9 Note that if there is such a point Θ^* in D which is a point of asymptotic stability for the mean ODE (14) with domain of attraction D, this means that any solution of (14) for $a \in D$ indefinitely remains in D and converges to Θ^* as $t \to \infty$. It can then be shown that (see [15], Th. 5.3, p.31) there exists a function $U(\Theta)$ which satisfies the conditions mentioned in Assumption 2.9.

Notation 2.2

$$\begin{aligned} K(c) &= \{\Theta;\ U(\Theta) \leq c\} \\ \tau(c) &= \inf(n;\ \Theta(n) \notin K(c)) \\ q_0(\alpha) &= \sup(2, 2(\alpha - 1)) \end{aligned} \tag{39}$$

With these notations and assumptions, we can present the following theorem (for a proof see [11], pp. 301-304):

Theorem 2.3 *Consider (21). Suppose Assumptions 2.1, 2.5, 2.6, 2.7, 2.8, 2.9 hold and suppose that F is a compact set such that*

$$F = \{\Theta;\, U(\Theta) \leq c_0\} \supset \{\Theta;\, U'(\Theta).h(\Theta) = 0\}$$

for some $c_0 < C$ where C is defined in Assumption 2.9. Then, for any compact subset Q of D, and $q \geq q_0(\alpha)$, there exists a constant $B_2(q)$ such that for all $a \in Q$:

$$P_a(\Theta(n)\ converges\ to\ F) \geq 1 - B_2(q) \sum_{i \geq 1} \epsilon_i^{1+\frac{q}{2}} \tag{40}$$

In the next subsection, we provide a simpler local bound for $q = 2$ following the analysis given in Section 5.1 of Part-II [11].

2.3 A simpler local bound for $q = 2$

Consider again the algorithm:

$$\Theta(n+1) = \Theta(n) + \epsilon_{n+1} H(\Theta(n), X_{n+1}),\ n \geq 0 \tag{41}$$

Since $X_n,\ n \geq 0$ are distributed independently of $\Theta(n)$ and also $\{X_n\},\ n \geq 0$ is a sequence of *i.i.d.* random variables, we have the main or so-called Robbins-Monro assumption satisfied, namely,

$$E[g(\Theta(n), X_{n+1}) \mid \mathcal{F}_n] = \int_{\mathbf{R}^d} g(\Theta(n), x) p(x) dx \tag{42}$$

Note that we have already observed before in Lemma 2.1 that

$$h(\Theta(n)) = E_a[H(\Theta(n), X_{n+1}) | \mathcal{F}_n] = \int_{\mathbf{R}^d} H(\Theta(n), x) p(x) dx \tag{43}$$

Next, we introduce the two main assumptions of this section:

Assumption 2.10 *For all $\Theta(0) = a \in \mathbb{R}^d$,*

$$E_a\left[|H(\Theta(n), X_{n+1})|^2 \mid \mathcal{F}_n\right] \leq \tilde{C}_1(1 + |\Theta(n)|^2) \tag{44}$$

for some suitable constant $\tilde{C}_1$.

Remark 2.10 Note that this assumption guarantees the existence of $h(\Theta(n))$.

Assumption 2.11 *$\exists \Theta^*$ (which is a point of asymptotic stability of (14) such that for all Θ, $\exists$ a constant $\delta > 0$ such that*

$$(\Theta - \Theta^*)' h(\Theta) \leq -\delta |\Theta - \Theta^*|^2 \tag{45}$$

with, for some $\beta \leq 1$,

$$\liminf_{n \to \infty} 2\delta \frac{\epsilon_n^\beta}{\epsilon_{n+1}} + \frac{\epsilon_{n+1}^\beta - \epsilon_n^\beta}{\epsilon_{n+1}^2} > 0 \tag{46}$$

Remark 2.11 Note that if $\epsilon_n = \frac{A_1}{n^\alpha + A_2},\ 0 \leq \alpha \leq 1$, then (46) holds for all $\beta < 1$. It is true for $\beta = 1$ if $2\delta > \frac{\alpha}{A_1}$.

We can now present the following theorem which gives a simple local bound for the expected distance between $\Theta(n)$ and Θ^*:

Theorem 2.4 *Consider (41). Suppose Assumptions 2.10, 2.11 hold. Then,*

$$E_a(|\Theta(n) - \Theta^*|^2) \leq B_5(a)\epsilon_n^\beta \tag{47}$$

for some suitable constant $B_5(a)$.

Proof: It is sufficient to show that for some suitable n_0, there exists a $B_5(a, n_0)$ such that for all $n \geq n_0$,

$$E_a(|\Theta(n) - \Theta^*|^2) \leq B_5(a, n_0)\epsilon_n^\beta \tag{48}$$

Writing $J_n \stackrel{\triangle}{=} \Theta(n) - \Theta^*$, we have

$$E_a(|J_{n+1}|^2 \mid \mathcal{F}_n) = |J_n|^2 + 2\epsilon_{n+1}\langle J_n, h(\Theta(n)\rangle + \epsilon_{n+1}^2 E_a[|H(\Theta(n), X_{n+1})|^2 \mid \mathcal{F}_n] \tag{49}$$

Suppose that n is sufficiently large such that $1 \geq 2\epsilon_{n+1}\delta$. Then, by taking expectations, we have

$$E_a|J_{n+1}|^2 \leq (1 - 2\epsilon_{n+1}\delta + \hat{C}_1\epsilon_{n+1}^2)E_a|J_n|^2 + \hat{C}_1\epsilon_{n+1}^2 \tag{50}$$

where $\hat{C}_1$ is a constant such that

$$\tilde{C}_1(1 + |\Theta|^2) \leq \hat{C}_1(1 + |\Theta - \Theta^*|^2) \tag{51}$$

Now, one can use the following result which can be proved directly from (46). There exists B^0 and n_0 such that for all $B_5 \geq B^0$ and $n \geq n_0$, the sequence $u_n = B_5\epsilon_n^\beta$ satisfies

$$u_{n+1} \geq (1 - 2\epsilon_{n+1}\delta + \hat{C}_1\epsilon_{n+1}^2)u_n + \hat{C}_1\epsilon_{n+1}^2 \tag{52}$$

Choose $B_5(a, n_0) \geq B^0$ such that

$$E_a|J_{n_0}|^2 \leq B_5(a, n_0)\epsilon_{n_0}^\beta$$

It follows immediately by induction on n that the sequence $u_n = B_5(a, n_0)\epsilon_n^\beta,\ n \geq n_0$ satisfies $E_a|J_n|^2 \leq u_n$ from which (47) follows. □

3 Simulation Studies

In this section, we present some simulation results illustrating the compression performance of our algorithm while a trade-off is obtained with respect to its classification performance. We consider two examples, one with computer simulated data distributed according to either of two bimodal Gaussian densities and the other with "mel-cepstral" coefficients of two female speakers obtained from their speech.

Bimodal Gaussian data

This part of the simulation study is carried out with computer generated random numbers distributed according to either of two two-dimensional bimodal Gaussian distributions. The first pattern is generated from the bimodal Gaussian density $0.5N([1.0\,1.0]', I) + 0.5N([-1.0\,-1.0]', I)$ where $N([m_1\,m_2]', \Sigma)$ is the two-dimensional normal distribution function with the mean vector $[m_1\,m_2]'$ and covariance matrix Σ. The second pattern is generated from the density $0.4N([0.0\,0.0]', 4I) + 0.6N([0.5\,0.5]', 4I)$. The training set was formed by 500 vectors from each pattern (meaning $\pi_1 = \pi_2 = 0.5$). This set was used to train the Voronoi vectors in multiple passes the total number of passes being 20. The number of Voronoi vectors that would result in a good classification performance was found by increasing the number of Voronoi vectors by one until the classification performance (for a given size of test data set) reached a floor. Thus 16 Voronoi cells were chosen and their centroids initialized by the output of an LBG algorithm processing the training data. Each test data set had a size of 1000 each containing vectors from pattern 1 and pattern 2 such that the a priori probabilities were satisfied. The learning rate ϵ_n was kept fixed over one pass such that $\epsilon_p = \frac{\epsilon_1}{\sqrt{p}}$ where p denotes the number of the pass, and $\epsilon_1 = 0.01$. The compression performance averaged over 10 test data sets for a range of $\lambda \in [0.0, 5.0]$ is given in Figure 1. The compression error was measured by the minimum mean square error that is the average of the squared distances between the test vectors and their representative Voronoi vectors and normalized with respect to the compression error achieved by the pure LVQ algorithm ($\lambda = 0.0$). It is seen that as λ increases up to 5.0, there is a reduction of approximately 3.5% in the normalized compression error.

The classification performance measured by the percentage of misclassified data did not change very much with increasing value of λ and tended to hover around 30% in the range of λ as mentioned above. Hence we did not include a separate plot for the classification performance.

Mel-Cepstral coefficients of 2 speakers

This example is based on "mel-cepstrum" coefficients of two female speakers. "Mel-cepstrum" features based on the nonlinear human perception of the frequency of sounds have been well studied and successfully applied to speaker identification problems. These studies have shown that the mel-cepstrum can effectively extract the vocal tract shape information of the speakers and yield good distinguishing performance [16] [17]. In our example, the labeled phonetic speech data of the two female speakers are extracted from the TIMIT database for dialect region 2. The speech waveform is segmented into 16 ms frames overlapped by 8 ms and parameterized to a 14 dimensional mel-cepstrum vectors to establish the feature space.

Since the performance of an LVQ type algorithm depends critically on the number of Voronoi vectors and the number of training vectors per Voronoi cell, to achieve a trade-off with the computational time required, the following parameters were chosen. The training set was randomly chosen to have 500 data vectors from each speaker. The number of Voronoi cells was chosen to be 20. The training set was used to update the Voronoi vectors in multiple passes, the total number of passes being 30. The learning rate

ϵ_n was taken to be constant over one pass where $\epsilon_p = \frac{\epsilon_1}{\sqrt{p}}$ where p denotes the number of passes with $\epsilon_1 = 0.04$. The Voronoi vectors were initialized by passing the training set through an LBG algorithm. Once the training was completed, 5 sets of test data, each containing 250 vectors taken randomly from the database for both speakers, were used to obtain the compression and classification performances of our algorithm. Figures 2 and 3 illustrate the results averaged over 5 test data sets, for a range of $\lambda \in [0.0, 5.0]$. As expected, the compression error (measured by the mean square distance between the data and its representative Voronoi vector), which was normalized with respect to the error obtained by the pure LVQ algorithm ($\lambda = 0.0$), decreases by approximately 7%, whereas the classification error goes up by 4.5%. We would like to comment here that the classification error can be further reduced by a choice of larger number Voronoi cells which would obviously require larger number of training vectors.

4 Conclusions and future research

We have developed an algorithm based on learning vector quantization (LVQ) for combined compression and classification. We have shown convergence of the algorithm, under reasonable conditions, using the ODE method of stochastic approximation. We have also illustrated the performance of the algorithm with some examples. The sensitivity of the performance of the algorithm with respect to the weight parameter λ indicates that the compression error decreases with increasing λ whereas the increase in classification error is relatively insignificant.

An important future research problem that we are currently working on is the extension of the algorithm when the VQ is replaced by TSVQ. In this extension, we use and extend the methods and analysis of [18]. With this extension, we will be able to treat the performance of the WTSVQ algorithm of [4] [5] [6], [7] analytically including compression of the wavelet coefficients.

References

[1] E. Frantzeskakis, *On Image Coding and Understanding: A Bayesian Formulation for the Problem of Template Matching Based on Coded Image Data*. 1990. M. S. Thesis, ISR Technical Report MS 90-5.

[2] K. O. Perlmutter, S. Perlmutter, R. Gray, R. Olsen, and K. L. Oehler, "Bayes risk weighted vector quantization with posteriror estimation for image compression and classification." preprint, 1997.

[3] K. L. Oehler and R. M. Gray, "Combining image compression and classification using vector quantization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, pp. 461–473, May 1995.

[4] J. Baras and S. Wolk, "Model based automatic target recognition from high range resolution radar returns," in *Proceedings of SPIE International Symposium on Intelligent Information Systems*, vol. 2234, (Orlando, FL), pp. 57–66, April 1994.

[5] J. Baras and S. Wolk, "Wavelet based progressive classification of high range resolution radar returns," in *Proceedings of SPIE International Symposium on Intelligent Information Systems*, vol. 2242, (Orlando, FL), pp. 967–977, April 1994.

[6] J. Baras and S. Wolk, "Wavelet based progressive classification with learning: Applications to radar signals," in *Proceedings of the SPIE 1995 International Symposium on Aerospace/ Defense Sensing and Dual-Use Photonics*, vol. 2491, (Orlando, FL), pp. 339–350, April 1995.

[7] J. Baras and S. Wolk, "Wavelet-based hierarchical organization of large image databases: ISAR and face recognition," in *Proceedings of SPIE 12th International Symposium on Aerospace, Defense Sensing, Simulation and Control*, vol. 3391, (Orlando, FL), pp. 546–558, April 1998.

[8] J. Baras and D. MacEnany, "Model-based ATR: Algorithms based on reduced target models, learning and probing," in *Proceedings of the Second ATR Systems and Technology Conference*, vol. 1, pp. 277–300, February 1992.

[9] D. MacEnany and J. Baras, "Scale-space polygonalization of target silhouettes and applications to model-based ATR," in *Proceedings of the Second ATR Systems and Technology Conference*, vol. 2, pp. 223–247, February 1992.

[10] A. LaVigna, *Nonparametric Classification Using Learning Vector Quantization*. PhD thesis, Dept. of Electr. Eng., University of Maryland, College Park, Maryland 20742, 1989. ISR Technical Report PhD 90-1.

[11] A. Benveniste, M. Metivier, and P. Priouret, *Adaptive Algorithms and Stochastic Approximation*. Berlin and New York: Springer-Verlag, 1990.

[12] J. S. Baras and A. LaVigna, "Convergence of a neural network classifier," in *Proc. of 29th IEEE Conf. on Decision and Control*, pp. 1735–1740, December 1990.

[13] T. Kohonen, *Self-Organizing Maps*. Heidelberg, Germany: Springer-Verlag, 1995.

[14] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Transactions on Communications*, vol. 28, pp. 84–95, 1980.

[15] Krasovskii, *Stability of Motion*. Stanford University Press, 1963.

[16] H. Gish and M. Schmidt, "Text-independent speaker identification," *IEEE Signal Processing Magazine*, pp. 18–32, October 1994.

[17] D. A. Reynolds, "Speaker identification and verification using gaussian mixture speaker models," *Speech Communication*, vol. 17, pp. 91–108, 1995.

[18] A. B. Nobel and R. A. Olshen, "Termination and continuity of greedy growing for tree-structured vector quantizers," *IEEE Transactions on Information Theory*, vol. 42, January 1996.

List of Figures

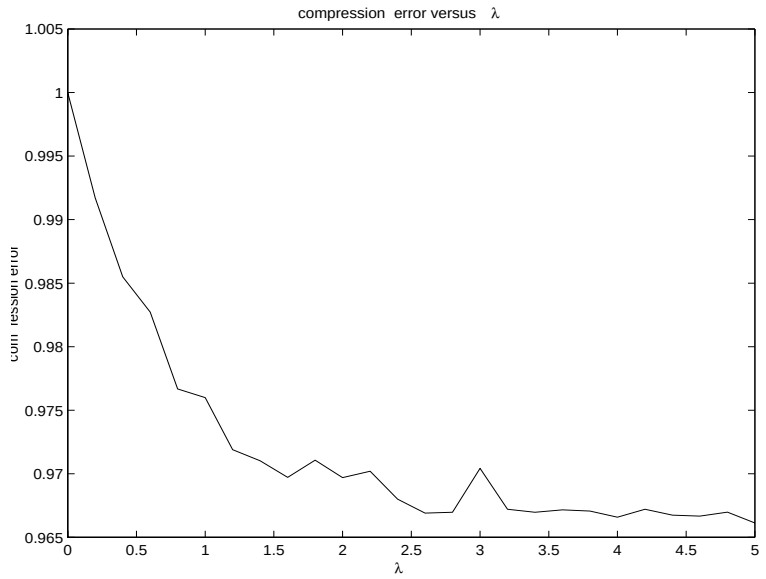

Figure 1: Compression error perfomance of the combined LVQ algorithm for bimodal Gaussian patterns

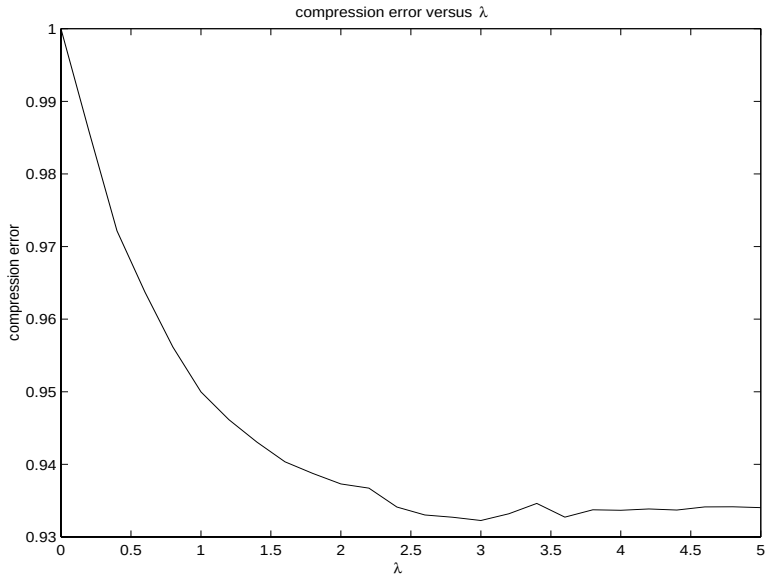

Figure 2: Compression error performance of the combined LVQ algorithm for “mel-cepstrum” features of female speakers

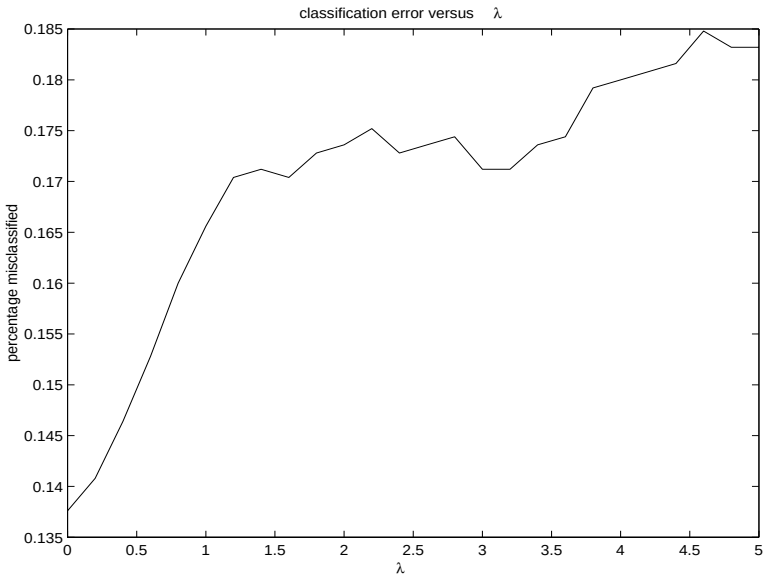

Figure 3: Classification error performance of the combined LVQ algorithm for “mel-cepstrum” features of female speakers